# Course 03/2

**Issues about accuracy and stability**

- Stability of Verlet algorithm
- Gear algorithm
- Performance of various algorithms

# Stability of Verlet algorithm

## Harmonic oscillator

$$\ddot{x}(t) = -\omega_0^2\, x(t)$$

Verlet evolution: $x(t+h) = 2\,x(t) - x(t-h) - h^2\,\omega_0^2\,x(t)$

Ansatz of solution on the discrete time steps: $x(t) = e^{\iota\omega t}$

Replacing in the Verlet evolution:

$$e^{\iota\omega(t+h)} = 2\,e^{\iota\omega t} - e^{\iota\omega(t-h)} - h^2\omega_0^2\,e^{\iota\omega t}$$

$$e^{\iota\omega h} = 2 - e^{-\iota\omega h} - h^2\omega_0^2$$

$$2 - 2\cos(\omega h) = h^2\,\omega_0^2$$

# Stability of Verlet algorithm

$$x(t) = e^{i\omega t} \longrightarrow 2 - 2\cos(\omega h) = h^2\omega_0^2$$

- If $h^2\omega_0^2 > 4$, there is no real solution for $\omega$, $\omega$ is imaginary

  $\rightarrow$ the solution diverges (easy to check!)

- If $\omega h \ll 1$, we expand the cosine:

$$\cancel{2} - 2\left[\cancel{1} - \frac{(\omega h)^2}{2} + \frac{(\omega h)^4}{4!} + \ldots\right] = h^2\omega_0^2$$

$$(h\omega)^2 \boxed{- \frac{1}{12}(h\omega)^4} = h^2\omega_0^2$$

Correction term depending on $h$ : $\lim_{h\to 0} \omega = \omega_0$

Existence of a "shadow Hamiltonian"

# Gear algorithm

General form of interest: $\ddot{x}(t) = f(t)$ (Newton equation)

Four-component quantity:

$$\vec{y}_n = \left( x_n, \ \Delta t\, \dot{x}_n, \ \tfrac{1}{2}\Delta t^2\, \ddot{x}_n, \ \tfrac{1}{6}\Delta t^3\, \dddot{x}_n \right)^T$$

1. Predictor step (Taylor expansion)

$$\vec{y}_{n+1}^{\,P} = \left( x_{n+1}^P, \ \Delta t\, \dot{x}_{n+1}^P, \ \tfrac{1}{2}\Delta t^2\, \ddot{x}_{n+1}^P, \ \tfrac{1}{6}\Delta t^3\, \dddot{x}_{n+1}^P \right)^T$$

$$\vec{y}_{n+1}^{\,P} = A\, \vec{y}_n \qquad \text{with } A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 \\ 0 & 0 & 1 & 3 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

2. Force calculation at predicted position $x_{n+1}^P$ :

$$f(x_{n+1}^P)$$

# Gear algorithm

## 3. Corrector step

$$\vec{y}_{n+1} = \vec{y}_{n+1}{}^{P} + \vec{a}\ \underbrace{\tfrac{1}{2}\Delta t^{2}[\, f(x_{n+1}{}^{P}) - \ddot{x}_{n+1}{}^{P}\,]}$$

4-component quantity

$$\vec{a} = \begin{pmatrix} 1/6 \\ 5/6 \\ 1 \\ 1/3 \end{pmatrix}$$

scalar quantity:
for the 3rd component of $\vec{y}_{n+1}$, the predicted acceleration $\ddot{x}_{n+1}{}^{P}$ is replaced with the one calculated at step 2, i.e. $f(x_{n+1}{}^{P})$.

- No iteration: just one force calculation.
- To optimize $\vec{a}$, Gear used criteria of accuracy and stability under the assumption that the force only depends on position, not on velocity.

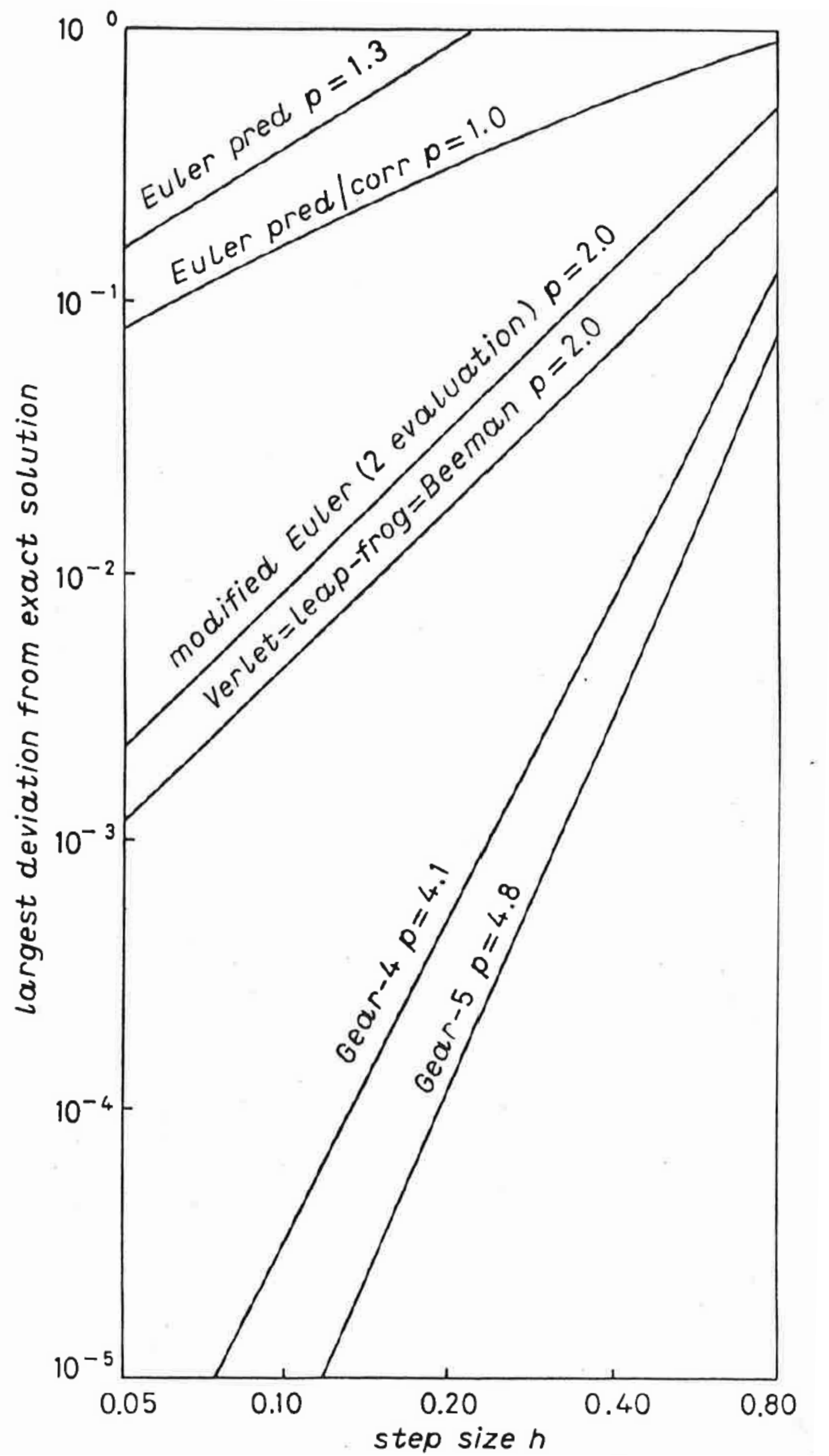
# Performance of various algorithms

Tests on two cycles of an harmonic oscillator

1st criterion:
Accuracy with respect to exact solution as a function of time step *h*

Overall accuracy

- Least accurate: Euler method.
- Most accurate: Gear method.



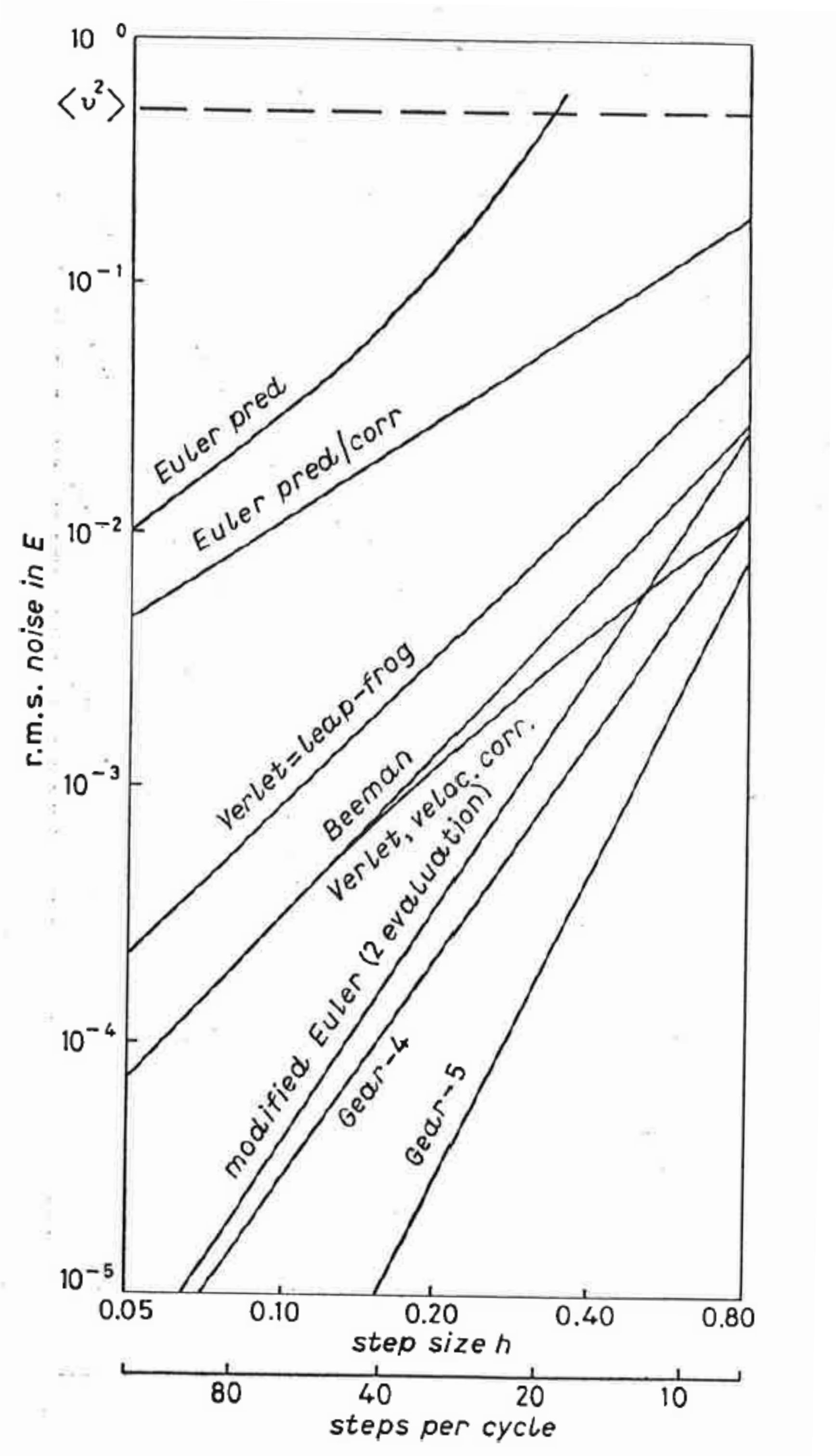
# Performance of various algorithms

Tests on two cycles of an harmonic oscillator

2nd criterion:
Root-mean-square deviation
from the linear regression line of
the energy as a function of time step *h*

Conserved energy (rms noise)

- Least accurate: Euler method.
- Most accurate: Gear method.
- Overall the same considerations as for the overall accuracy, but Gear and Verlet appear now to be comparable at large time steps.



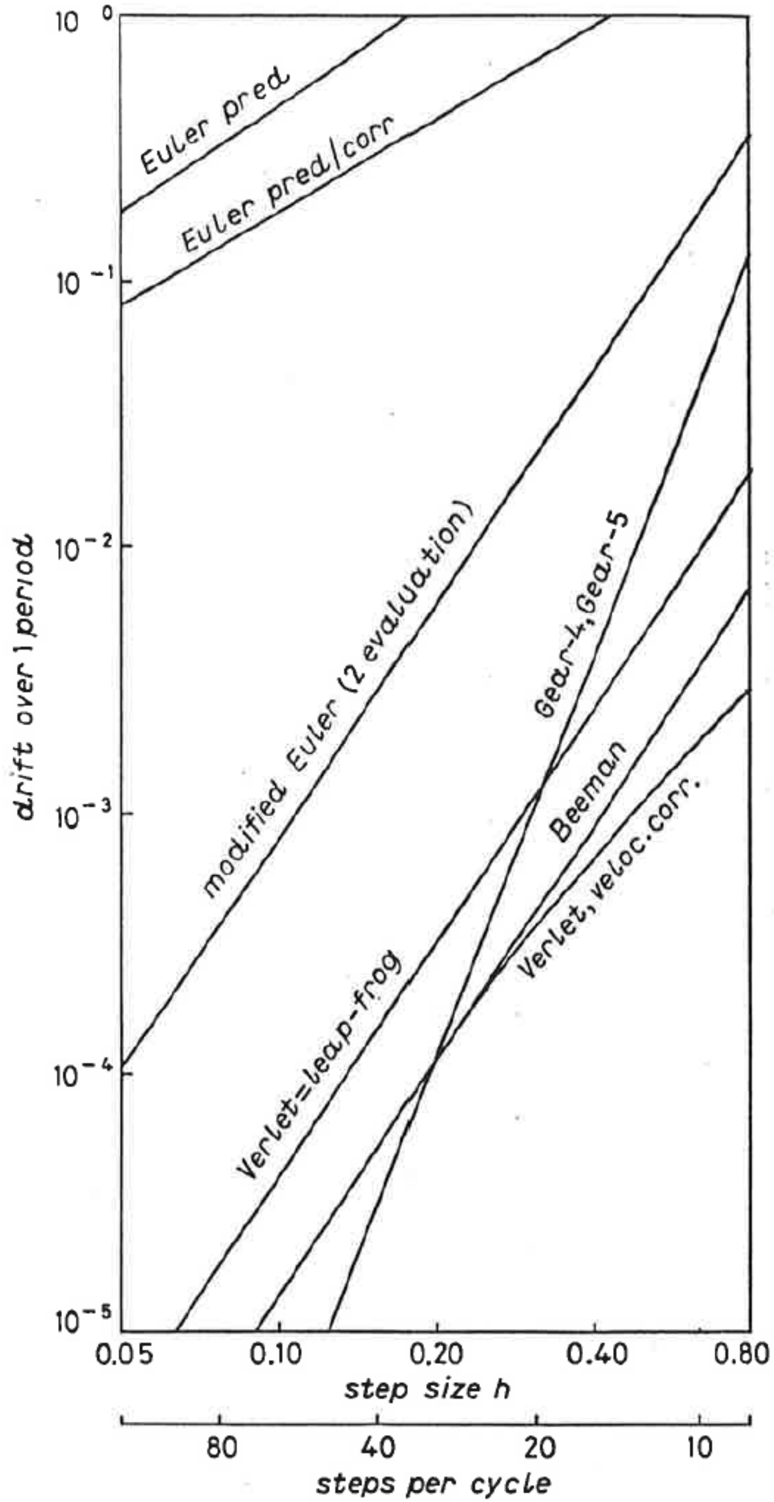
# Performance of various algorithms

Tests on two cycles of an harmonic oscillator

3rd criterion:
Drift of the line of energy line
as a function of time step *h*

Drift in the energy

- Verlet beats Gear for large time steps.
- For long simulation, Gear algorithms are less efficient.



# Course 03/2

**Issues about accuracy and stability**

- Stability of Verlet algorithm
- Gear algorithm
- Performance of various algorithms